Лабораторная работа №6

Тема «Исследование асимметричных криптоалгоритмов»

ВЫПОЛНИЛ

Олишкевич Игорь Русланович

Группа 25-ПО

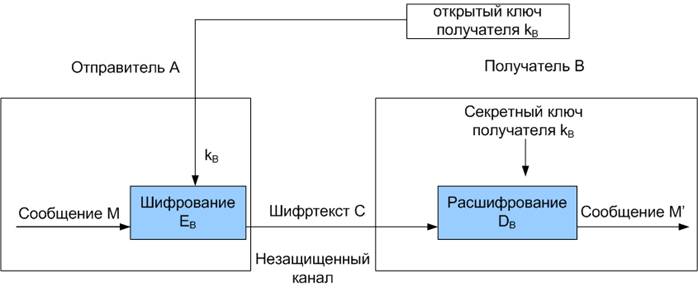
Задание: изучить асимметричный шифр; привести примеры, схему.

Асимметричные алгоритмы шифрования (или алгоритмы с открытым ключом) – криптографические алгоритмы, в которых один ключ используется для шифрования, а другой, отличный от первого, – для расшифрования. Алгоритмы называются асимметричными, так как ключи шифрования и расшифрования разные, следовательно, отсутствует симметрия основных криптографических процессов. Один из двух ключей является открытым (public key) и может быть объявлен всем, а второй – закрытым (private key) и должен держаться в секрете. Какой из ключей, открытый или закрытый, используется для шифрования, а какой для расшифрования, определяется назначением криптографической системы.

Все алгоритмы шифрования с открытым ключом основаны на использовании односторонних функций. Односторонней функцией называется математическая функция, которую относительно легко вычислить, но трудно найти по значению функции соответствующее значение аргумента. Использовать односторонние функции для шифрования сообщений с целью их защиты не имеет смысла, так как обратно расшифровать зашифрованное сообщение уже не получится. Для целей шифрования используются односторонние функции с люком (или с секретом) – особый вид односторонних функций, имеющих некоторый секрет (люк), позволяющий относительно быстро вычислить обратное значение функции.

Алгоритмы шифрования с открытым ключом можно использовать для решения следующих задач:

1. для шифрования передаваемых и хранимых данных в целях их защиты от несанкционированного доступа;
2. для формирования цифровой подписи под электронными документами;
3. для распределения секретных ключей, используемых потом при шифровании документов симметричными методами.

  
Рисунок 1 –Схема ассиметричных криптоалгоритмов

Асимметричные криптосистемы обладают неоспоримым достоинством по сравнению с симметричными: они позволяют динамически передавать открытые ключи, тогда как при использовании симметричной криптосистемы необходим обмен секретными ключами до начала сеанса защищенной связи. Асимметричные криптоалгоритмы позволяют преодолеть недостатки, присущие системам симметричного шифрования:

* + не требуется секретная доставка ключей;
  + исчезает квадратическая зависимость числа ключей от числа пользователей.

Недостатки асимметричных криптосистем:

* + на настоящий момент нет математического доказательства необратимости используемых в асимметричных алгоритмах однонаправленных функций;
  + по сравнению с симметричным асимметричное шифрование существенно медленнее, поскольку при зашифровании и расшифровании выполняются весьма трудоемкие операции (в частности, в RSA это возведение одного большого числа в степень, являющуюся другим большим числом). По этой же причине реализация аппаратного шифратора с асимметричным криптоалгоритмом существенно сложнее, чем аппаратная реализация симметричного криптоалгоритма;
  + необходимо защищать открытые ключи от подмены.

Алгоритм RSA стоит у истоков асимметричной криптографии. Он был предложен тремя исседователями-математиками Рональдом Ривестом (R.Rivest), Ади Шамиром (A.Shamir) и Леонардом Адльманом (L.Adleman) в 1977-78 годах.

Первым этапом любого асимметричного алгоритма является создание пары ключей: открытого и закрытого и распространение открытого ключа "по всему миру". Для алгоритма RSA этап создания ключей состоит из следующих операций:

1. выбираются два простых (!) числа p и q;
2. вычисляется их произведение n(=p\*q);
3. выбирается произвольное число e (e<n), такое, что НОД(e,(p-1)(q-1))=1, то есть e должно быть взаимно простым с числом (p-1)(q-1);
4. методом Евклида решается в целых числах (!) уравнение e\*d+(p-1)(q-1)\*y=1. Здесь неизвестными являются переменные d и y – метод Евклида как раз и находит множество пар (d,y), каждая из которых является решением уравнения в целых числах;
5. два числа (e,n) – публикуются как открытый ключ;
6. число d хранится в строжайшем секрете – это и есть закрытый ключ, который позволит читать все послания, зашифрованные с помощью пары чисел (e,n).

Шифрование с помощью этих чисел:

1. отправитель разбивает свое сообщение на блоки, равные k=[log2(n)] бит, где квадратные скобки обозначают взятие целой части от дробного числа;
2. подобный блок может быть интерпретирован как число из диапазона (0;2k-1). Для каждого такого числа (mi) вычисляется выражение ci=((mi)e)mod n. Блоки ci и есть зашифрованное сообщение. Их можно спокойно передавать по открытому каналу, поскольку.операция возведения в степень по модулю простого числа, является необратимой математической задачей. Обратная ей задача носит название "логарифмирование в конечном поле" и является на несколько порядков более сложной задачей. То есть даже если злоумышленник знает числа e и n, то по ci прочесть исходные сообщения mi он не может никак, кроме как полным перебором mi.

На приемной стороне процесс дешифрования возможен с помощью хранимого числа d. Достаточно давно была доказана теорема Эйлера, частный случай которой утвержает, что если число n представимо в виде двух простых чисел p и q, то для любого x имеет место равенство (x(p-1)(q-1))mod n = 1. Для дешифрования RSA-сообщений воспользуемся данной формулой. Возведем обе ее части в степень (-y): (x(-y)(p-1)(q-1))mod n = 1(-y) = 1. Теперь умножим обе ее части на x: (x(-y)(p-1)(q-1)+1)mod n = 1\*x = x.

На самом деле операции возведения в степень больших чисел достаточно трудоемки для современных процессоров, даже если они производятся по оптимизированным по времени алгоритмам. Поэтому обычно весь текст сообщения кодируется обычным блочным шифром (намного более быстрым), но с использованием ключа сеанса, а вот сам ключ сеанса шифруется как раз асимметричным алгоритмом с помощью открытого ключа получателя и помещается в начало файла.

Проблема защиты открытых ключей от подмены: предположим, на компьютере абонента А хранится открытый ключ КВ абонента В. Пусть злоумышленник @ имеет доступ к открытым ключам, хранящимся у абонента А. Он генерирует свою пару ключей k@ и К@ и подменяет у абонента А открытый ключ КВ абонента В на свой открытый ключ К@. Для того чтобы отправить некую информацию абоненту В, абонент А зашифровывает ее на ключе К@, думая, что это ключ КВ. Соответственно, это сообщение не сможет прочитать абонент В, но зато легко расшифрует и прочитает злоумышленник @. Атаку подмены открытых ключей может предотвратить процедура сертификации открытых ключей.